MODIS
SCIENCE DATA SUPPORT TEAM
PRESENTATION

June 5, 1992

AGENDA

Page

ACTION ITEMS:

04/24/92 [Lloyd Carpenter] Prepare the Team Leader's Software and Data Management Plan for review. (The latest draft version was distributed at the meeting on May 8, 1992 and provided to the Miami TMs on 5/20.) STATUS: Open. Due Date: May 10, 1992.

04/24/92 [Lloyd Carpenter] Prepare the Team Leader's Science Computing Facility Plan for review. (The latest draft version was distributed along with the handout at the May 1st meeting and provided to the Miami TMs on 5/20.) STATUS: Open. Due Date: May 10, 1992.

04/24/92 [Tom Goff] Develop a detailed schedule through to the delivery of Version 1 to the DAAC for Level-1A and -1B software design and development, identification of risk areas in Level-1A and -1B design, and prototyping of risks. (A draft task list and schedule are included in the handout.) STATUS: Open. Due Date: 05/22/92

04/24/92 [J. J. Pan] Develop a detailed schedule for the Level-2 Processing Shell design and development, identification of risk areas in the Level-2 Processing Shell design and development, and prototyping of risks, through to the delivery of Version 1 to the DAAC. (An extensive outline and schedule were included in the handout on May 29, 1992. A draft Risk Analysis is included in today's handout.) STATUS: Open. Due Date: 05/22/92

04/24/92 [J. J. Pan] Develop a detailed schedule for a typical algorithm integration into the Level-2 processing shell. (A draft task list and schedule are included in the handout.) STATUS: Open. Due Date: 06/05/92

04/24/92 [Lloyd Carpenter & Team] Develop a staffing plan for the accomplishment of the tasks shown on the schedule. STATUS: Open. Due Date: 06/12/92

DRAFT VERSION of the
Work Task List
for the
MODIS Level-1A and 1B
Data Product Generator Designs

Thomas E. Goff
4 June, 1992

INTRODUCTION: This document contains a list of task items to be performed in order to create the MODIS Data Product Generator (DPG) programs that will produce the MODIS Level-1A and Level-1B Data Products. This list includes items not directly applicable to the design effort but which must be accomplished in order to assure a timely and professional execution of the MODIS DPG programs. These items include Configuration Management (CM), prototyping, performance profiling, key index database generation, and training of personnel. A Ghant chart of the first 2½ years of this multi-year effort is included.

This list is limited to the development and testing of the β (beta) delivery of the software, and the development only of the revision 1 version of the software. It is to be used for manpower estimates for the early stages of development and the synchronization of this level of effort with external levels of effort. Estimates for the testing of the version 1 and subsequent efforts will be added at a later date.

NOTES: dependencies are given in curly braces {}, one person duration (manpower) times are given in brackets [], and comments are given in *italics*.

1. Software Requirements Specification [3 weeks]
   Provide a full description to SDST specs - combining appropriate elements of the IEEE and NASA requirements specification standard guidelines.

2. Data Rate & Volume Spreadsheet {Excel spread sheet} [1 week]
   Clean up and greatly expand the information in the existing or a newly created spread sheet to include a full description of all quantities. The emphasis will be on clarity. This will be used as an input to the EIS writeup and to determine the DPG data input requirements.

3. CASE design {cadre's teamwork, training}
   Perform the next revision of the design in a full CASE environment. This will be an on-going process throughout the project life span (15+ years). The major CASE subitems given below are given up to the β (beta) and $\gamma 1$ (version 1) releases of the software. Both the Level-1A and Level-1B designs are included.

3.1 β Structure Charts [1 week per bubble, 38 bubbles, concurrent]
   For each processing element, using the EasyCASE structure chart bubble levels, as derived from the existing preliminary design. The next revision of the design will be performed using Cadre's TeamWork on the TLCF UNIX computer. This effort will expanded the design to the source code level. A List of the processing bubbles follows:

Level-1A DPG

3.1.1 Control Message Decomposition
3.1.2 Abort / Cleanup Determination
3.1.3 Processing Mode Setup
3.1.4 Status Derivation

3.1.5 Data Availability Check
3.1.6 Memory Requirement Determination
3.1.7 Memory Request
3.1.8 Data Store Setup
3.1.9 Packet Parameter Determination
3.1.10 Packet Structure Verification
3.1.11 Time Stamping / Formatting
3.1.12 Packet to Scan Cube Location
3.1.13 Scan Vector Direction Determination
3.1.14 Attitude Knowledge Correction
3.1.15 Find Earth Intersection
3.1.16 Find Moon Intersection
3.1.17 Convert to Lat/Long System
3.1.18 Post Anchor Points
3.1.19 Granule virtual Translation
3.1.20 Completeness Indicator Updating
3.1.21 Scan Cube Completeness Determination
3.1.22 Ancillary Data Appending
3.1.23 Cube Header Creation
3.1.24 Metadata Updating
3.1.25 Granule completeness Determination
3.1.26 Metadata Derivation
3.1.27 Granule Header Appending
3.1.28 Granule and Header Transmission

Level-1B DPG

3.1.29 Verify Level-1A input to Level-1B DPG
3.1.30 Unpack the data (this function may be placed in Level-1A design)
3.1.31 Obtain the calibration Sensor values
3.1.32 Detect any calibration anomalies
3.1.33 Apply calibration curves
3.1.34 Determine the data quality
3.1.35 Replace raw counts with calibrated radiances in the scan cube.
3.1.36 Append Level-1B Metadata
3.1.37 Update Level-1B header
3.1.38 transmit Level-1B data to DADS

3.2 Data Dictionary {Teamwork} [concurrent with the structure charts]

This will be performed as a necessary component of the structure chart design and will continue to be updated for the duration of the project. The data dictionary will contain the definitions of all processing elements, external entities, and data structures. A separate assumptions description and item tracking list will be maintained.

3.2.1 Data Structure Layout {Teamwork, FrameMaker}

These are the data arrays that will be ingested or generated by the Level-1A process. They consist of instrument data in addition to the contents of the various internal and external messages.

3.2.1.1 Level-0 Packet contents {data from SBRC} [2 weeks]

These will be delineated by packet ID (calibration vs science day/night modes, etc)

3.2.1.2 DADS Interrogation Messages {PGS tool kit} [4 weeks]

These messages allow the MODIS DPG to query the DADS to determine the availability of the input data required to perform the program execution. Returned from the DADS will be the location of the input data sets from which the DPG can determine the advisability of proceeding with the processing. The DADS will then have the opportunity of staging the data locally if needed.

3.2.1.3 Level-1a Data Product contents {Level-0 Packet contents} [1 week]
This is the structure of the packets of MODIS data from a programming viewpoint. This is a formal data structure.

3.2.1.4 Metadata contents [4 weeks]
This task is distributed across the design effort to allow for evolving additions, deletions, and clarification of metadata items. This task item includes the development time up to the delivery of the β version of the software. Additional development time will be needed for the follow-on revisions.

3.2.1.5 Cube and Granule Header contents {SDST agreement} [1 week]
This is the formal data structure (format) of the scan cube and the cube header for both the Level-1A and Level-1B Data Products. Note that the header is a superset of the metadata.

3.2.1.6 Initiation Message {PGS tool kit} [1 week]
This item defines the content and structure (format) of the process initiation messages from the SCA.

3.2.1.7 Termination Message {PGS tool kit} [1 week]
This item defines the content and structure (format) of the process termination messages to the SCA.

3.2.1.8 Dynamic Status Inquiry/Response {PGS tool kit} [1 week]
These are the messages that are sent either solicited or unsolicited between the DPG and the SCA.

3.2.1.9 Processing Log entries {on going} [1 week]
This data structure defines the format and content of messages will be posted to the MODIS processing log and/or also the PGS processing log. Hopefully the message structure will be the same if both entities are to receive them.

3.2.2 Data Flow [2 weeks]
These items of flow control are defined in the data dictionary, formatted into messages and passed to and received from external processes.

3.2.2.1 Instrument Events {MCST definitions}
This message will have a variable content that will delineate any events or problems with the MODIS instrument as detected by the MODIS DPG.

3.2.2.2 Processing Events
This item will generate processing problems of a critical or uncritical nature that can be sent to interested external entities.

3.2.2.3 System Memory Allocations {PGS tool kit}
This delineates the call parameters and sequence that allow the MODIS DPG to allocate and receive computer memory necessary to perform the MODIS processing.

3.2.3 Processing Items [2 weeks]
These items must be derived as part of the data dictionary effort. They are not covered by the

normal data processing bubbles.

3.2.3.1 Initialization and Setup of Metadata and Granule Outlines
These are the internal data structures for the metadata and MODIS instrument data.

3.2.3.2 Granule Virtual Paging
This determines the form of the internal MODIS, virtual paging, data structure to be used to determine demand paging criteria (dirty bits).

3.3 Version 1 (γ1) structure charts updates. {β version} [84 weeks]
This is the continuation of the CASE structured environment to cover the version 1 release of the MODIS DPG software. It covers the development debugging, but not the PGS testing. The remaining effort can not be well defined until the ECS computer system has been specified.

4. Assumptions / Tracking List {on going} [concurrent]
A continuing and expanding effort that is kept concurrent with the design effort.

5. Coding {TLCF computer / SoftBench}
The generation of computer source code in an environment that is coupled to the structured design charts. Integrated debugging of code and configuration management of code and complementary files will be a part of this effort.

5.1 Source File Creation/Debugging {TLCF compilers} [30 weeks]
Perform the actual code generation, compiling, and debugging according to the SDST coding guidelines.

5.2 MakeFiles {Make manuals} [2 weeks]
Creating (if not performed automatically by SoftBench) of the interdependencies of all the modules that constitute the executing programs and on-line documentation generation.

5.3 Code QA Checking (tools) {aux computers} [3 weeks]
Checking of all source code against predetermined quality assessment (QA) criteria. This is expected to be an automatic procedure that will be accessible to the SDST but will probably reside on EOS Project or other (than the TLCF) computers.

5.4 Code walk throughs {on going} [12 weeks]
This is the hand inspection of all critical code at a very detailed level by knowledgeable software engineers. This is <u>not</u> a code style exercise.

5.5 Source Code Feature Extraction {TLCF PERL/ scripting languages} [2-5 weeks]
This consists of scripts or programs that will extract on-line documentation from source code files and perform a continuous updating of these information files. This is expected to be performed automatically, ultimately as part of the Make facility.

5.6 Key Index Database {TLCF computer} [5 weeks]
This program or script will allow all users of the computer system to find modules, data structure descriptions, and other programming information quickly using automated procedures. This will allow the maximum reuse of code and structures.

6. External Interface Document {PGS tool kit} [2 weeks]
This document will describe and detail the interactions among the MODIS Level-1A processor and

the external (to the MODIS Level-1A processor) environment. The external entities are listed below:

6.1 Wall Time
6.2 DTM
6.3 DADS
6.4 PMS
6.5 SCA

7. Prototyping
This effort provides early derivation and testing of selected portions of the design. Items generated during this prototyping effort will be incorporated into the final design.

7.1 IFOV Ground Plots {Scan Geometry}
Graphical plots that illustrate the MODIS IFOV coverage on differing geographically based maps will allow researchers to visualize the problems associated with IFOV overlap due to Earth curvature and the instrument scan 'bow tie' effect.

7.1.1 Plotting Package {library} [1 week]
A generalized plotting package will be installed on the TLCF. This will probably be the gnu plotting package or equivalent, with minor enhancements added.

7.1.2 Projection Routines [4 weeks]
This allows translation from WGS84 lat-longs to a flat 2-D map.

7.1.3 MODIS Scan Geometry implementation {SBRC / MCST} [4 weeks]
Coding and numerical validation of the MODIS scan geometry equations.

8. Scan Geometry derivation [4 weeks]
Obtaining / generation of the equations to convert from instrument geometry parameters to Earth intersection via IFOV vectors.

8.1 Anchor Point Determination {SBRC geometry}
Deriving or otherwise obtaining the MODIS scan geometry equations and converting this information into instrument pointing vectors.

8.1.1 Earth Model {PGS Tool Kit, Earth model decision}
The analytical representation of the Earth geoid.

8.2 Moon Looking {flight dynamics code and/or algorithms}
The analytical equations and data that can be used to determine if the Moon is in the instrument FOV and upon which pixels the Moon image is projected.

9. PGS Tool Kit Interfacing {ECS contractor Tool Kit Spec Delivery}
Included in this topic are items that will be necessary to complete the MODIS Level-1A processor design and/or verification. This includes, but is not limited to, the following items:

9.1 S/C Ephemeris Simulator {flight dynamics / homegrown}
To provide S/C platform position and attitude values at either the S/C telemetry packet intervals, or at the MODIS requested UTC time.

9.2 Digital Terrain Model, database

Terrain Elevation and possibly Slope at user requested positions. This is to be provided in several, user selectable, coordinate systems.

10. Testing/Debugging/profiling {computer} [30 weeks]
    Normal debugging will be handled within the CASE / SoftBench environment. Auxiliary items not available from the CASE environment are listed below:

10.1 Packet Telemetry Simulator {SBRC telemetry lists} [8 weeks]
GTSIM is a real-time simulator and needs to be checked for possible use for MODIS. Otherwise, a telemetry simulator for MODIS that can generate staged telemetry via the (simulated) DADS needs to be developed.

10.2 L-1A Data Product Validator/Invertor {Level-1A design} [8 weeks]
This utility item receives the Level-1A Scan Cube and verifies the contents by any of various analytical or visual techniques. This may contain COTS image processing facilities.

The Level-1A to Level-0 data invertor will be used recursively with the MODIS Level-1A Data Product Generator to validate the Level-1A Data Products. This invertor should be written by an independent validation authority.

10.3 DPG Performance Determination on TLCF computer {Level-1A design) [4 weeks]

Using the TLCF profiler, determine where the computer is spending its time during the execution of the Level-1A Data Product Generator. This will give an indication of the performance of the final Level-1A DPG as installed on a similar computer. These results can be extrapolated to the final PGS computer when available, being more accurate if the PGS computer is closer architecturally to the TLCF.

11. Documentation {FrameMaker or equivalent on TLCF}
    This allows the generation and electronic dissemination of all documents. The documents can also be worked on by more than one person simultaneously.

11.1 EIS for MODIS {MODIS spread sheet} [3 weeks]
This is best written on the TLCF, but will probably be written on PC or Mac based word processors and imported into FrameMaker when it becomes available.

11.2 Software Design Description {CASE design} [3 weeks]
A written description of the software design with requirements and assumptions included. This accompanies the structured design charts and data dictionary to form a complete design document.

11.3 Software V&V Plan {Code Review unless performed independently} [8 weeks]
This includes the writing of the V&V plan and the presentation of the results of executing the V&V plan.

11.3.1 Traceability Matrix {CASE design}
This document correlates the elements of the design with the design requirements. This is performed in two directions: from the requirements to the structured design and from the design to the requirements.

11.4 User {delivered with code releases} [2 weeks]
This is the User's written guide, where the user will be the PGS implementers and operators.

12. Configuration Management {CM tool / networked} [2 weeks]

Exercising the CM tool, hopefully integrated into the SoftBench environment. This may require some setup in order to allow computerized automation if a CM tool is selected after coding has begun.

13. Training

This is a list of formal training that needs to be provided to allow optimal efficiency of the development staff.

13.1 formal TLCF computer classes [4 weeks]

Vendor supplied classes in using the computer, learning the internal operation of UNIX, and system administration.

13.2 Cadre's Teamwork Usage Classes [2 weeks]

Formal training at Cadre and informal (at TLCF) training.

13.3 SoftBench Usage [2 weeks]

Vendor supplied training in the use and installation of SoftBench.

13.4 Configuration Management Tool [2 weeks]

Vendor supplied training in the selected CM tool. This selection should also be performed very soon after coding in initiated.

13.5 Programming QA Tools [2 weeks]

Training (possibly informal) in the use of all available code QA tools.

RISKS:

* Direct communication with SBRC.

  Channels need to be created between the SDST and SBRC for direct access to instrument information.

* TLCF availability of the complete computer system.
  - System Administration.
  - Vendor support for hardware maintenance, software updates, and consulting.

* TLCF computer capabilities

  The TLCF computer operation system needs to be able to perform preemptive multitasking of executing processes to allow the passing of control flow messages among processes.

* Timeliness of Teamwork training.

  This is the first training that needs to be completed.

* Timely selection of all tools that are to be integrated in to the SoftBench environment.

  This includes a configuration management tool with integrated Make facility, documentation publishing tools and printers, and any SDST (or other party) developed tools such as the key index data base. Data analysis tools that are not a part of the SoftBench environment such as image processing and data visualization tools can also be included in this risk item.

* Verification of packet simulator.

A bit by bit comparison with explained discrepancies of the simulated packets against SBRC supplied packets.

* Verification of the Level-1A validator.
Perhaps SBRC generated test data can be used?

* The Independence of the Validation Team.
The validation should ideally be performed in a 'clean room' environment by independent institutions. This may not be possible given the amount of work to be performed.

* PGS tool kit specifications.
Early information about the PGS Tool Kit will allow this MODIS design effort to avoid delays. This is especially important in the interfacing with the DADS and other low level operating system interfaces.

* Timeliness and thoroughness of the programming style reviews and code walk throughs.
This effort requires a comprehensive review of code performed within a minimum turn around.

* The possible production of separate Level-1B products for each of the land, ocean, and atmospheric communities.
If these disciplines require differing calibration techniques, then the possibility of differing Level-1B Data Products exists.

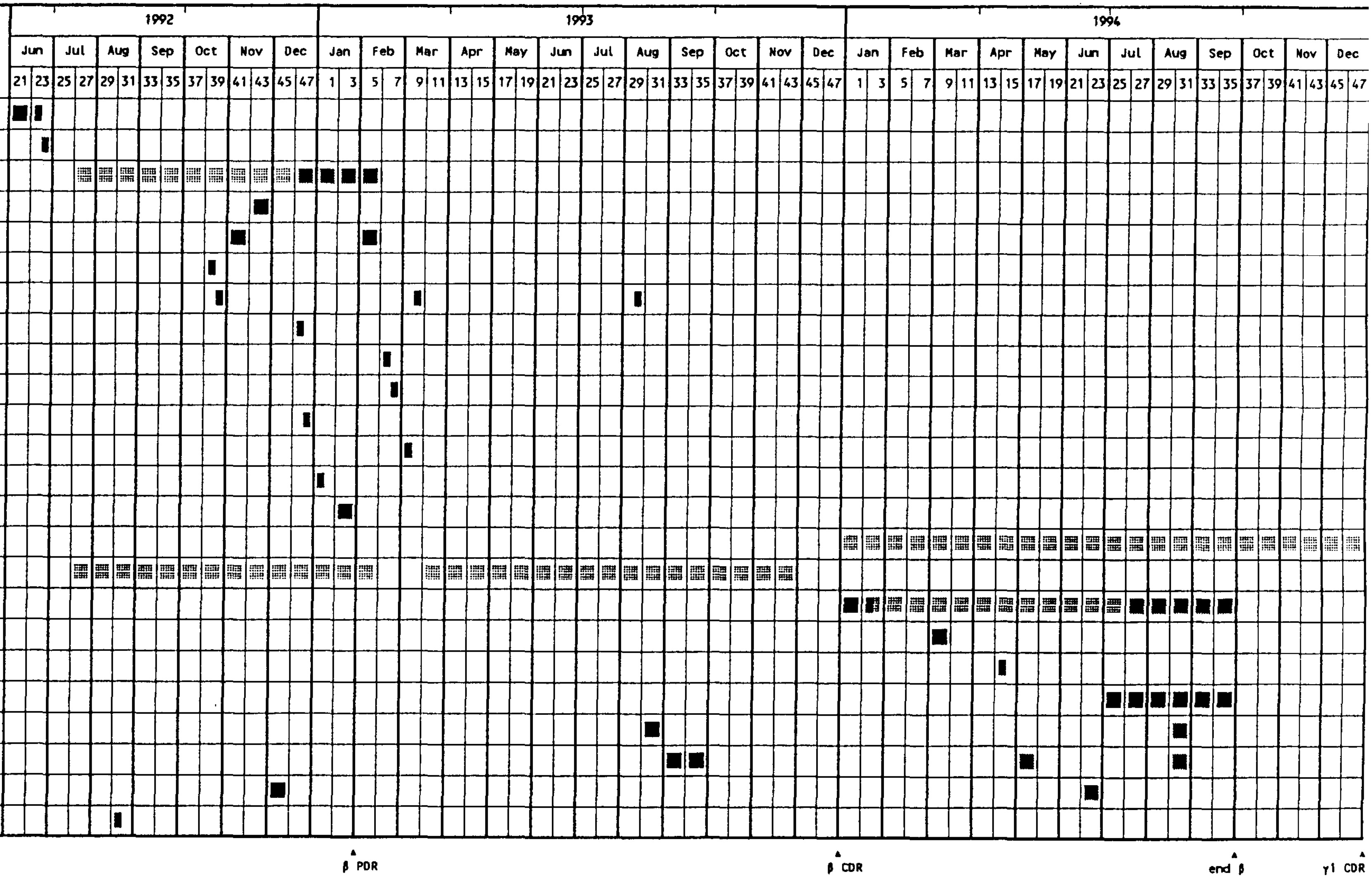
file: c:\modis\task_l1.wp

4 June, 1992

MODIS Science Data Support Team (SDST) Schedule for the Level-1A and Level-1B Data Product Generators

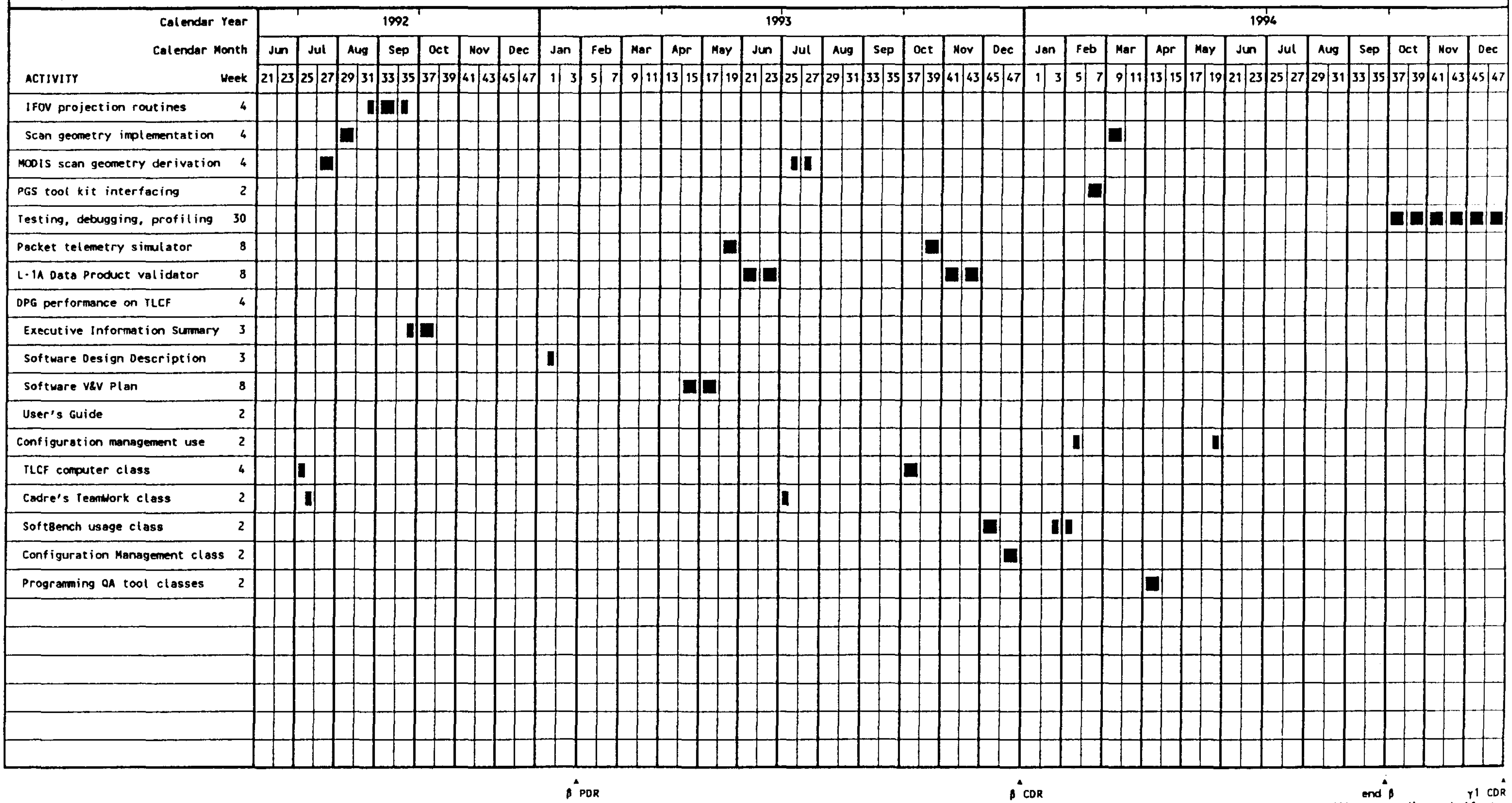
Thomas E. Goff

ACTIVITY	Week
Software Requirements Spec	3
Data rate & volume spreadsheet	1
Develope β structure charts	28
Level-0 packet contents	2
DADS interrogation messages	4
Level-1A Data Product contents	1
Metadata contents	4
Cube & granule header contents	1
Initiation messages	1
Termination messages	1
Dynamic status interrogations	1
Processing log entries	1
Data flow events	2
Processing item setup	2
Version 1 structure charts	10
Assumptions/tracking list	
Source code creation & debug	30
Makefiles	2
Source code QA, automated	3
Code walkthroughs	12
Source code feature extraction	4
Key index database	5
External Interface Document	2
Generalized plotting package	1



4 June, 1992

MODIS Science Data Support Team (SDST) Schedule (continued)



file: c:\modis\task_l1.gtc

Schedule for A Typical Algorithm Integration into the Level-2 Processing Shell

J. J. Pan
Research and Data Systems Corp.
(301) 982-3700

The detailed schedule developed here is based on the assumption that some tools in the PGS Toolkit, particularly the Level 1-4 Product Access Tools, Temporary File I/O Tools, Browse Output Tools, and Quicklook Product Tools, are available before April, 1993. Also it is expected that a typical algorithm for testing is available before May, 1993. The task includes the following activities:

1. Science Team Member Code Development
   -- It is expected that some TM will deliver their code before PDR, so that we can select a typical algorithm for the integration test. The SDST will contact with TM frequently and suggest a typical algorithm which can explore the accuracy of system concept.

2. Test of A Typical Algorithm
   -- A typical algorithm will be selected. Check the source code, installation instructions, operating instructions, and other documentation. Use the test data provided by TM to test all paths in the code, examine the output data sets, and communicate with TM for any inconsistencies. The iterative process between SDST and TM will continue until the performance of the algorithm is correct.

3. Shell Development
   -- A preliminary shell development will start in April, 1993, particularly with emphasis on how to manipulate input data efficiently and keep the shell robust and highly flexible for TM code modification. Also, document the interface between the algorithm and the shell. An intensive communication between SDST and PGS/ECS staff is necessary for the specifications of the Toolkit requirements.

4. PGS Toolkit Development
   -- The PGS Toolkit may not be available in April, 1993. However, it will be helpful to set the priority of required tools, so that ECS contractor can deliver these highly required tools to PGS and TLCF at an early stage. The major schedule risk is that the final specifications review of the Toolkit is scheduled at the 4th quarter, 1993.

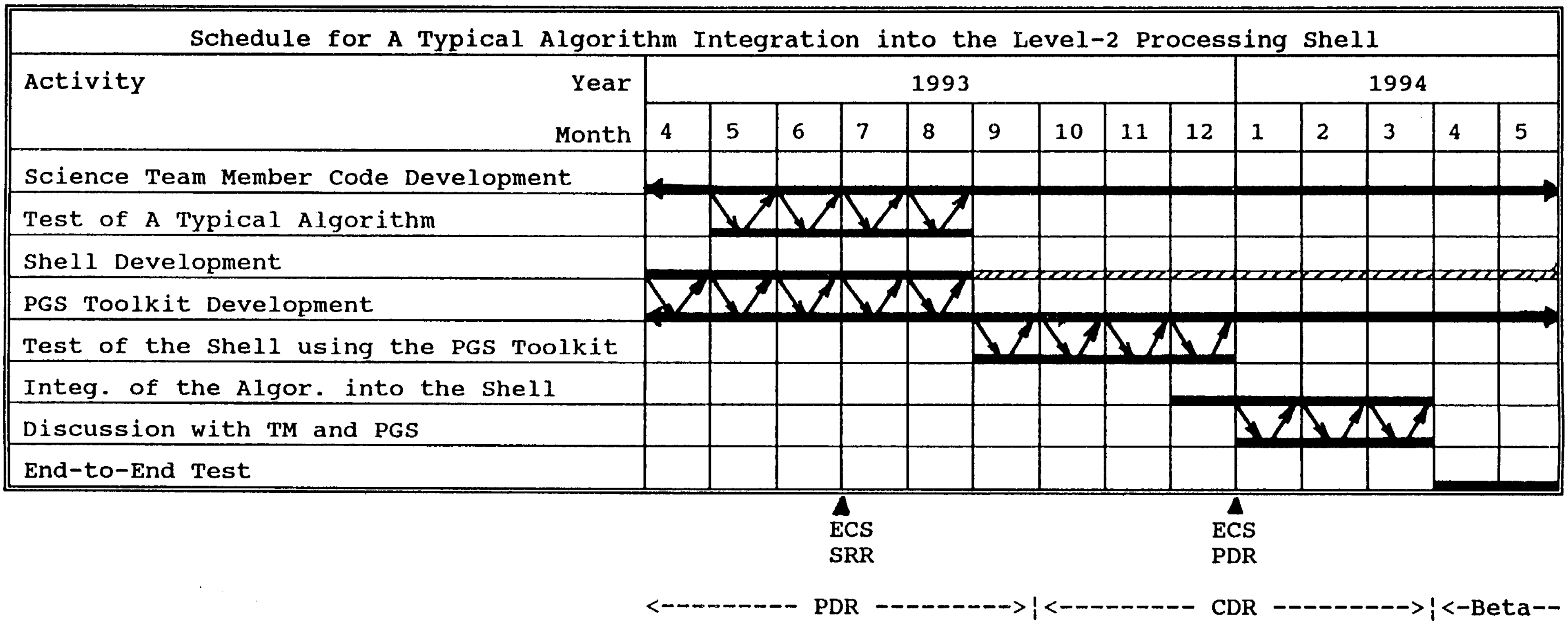
5. Test of the Shell Using the PGS Toolkit
   -- It is expected that required tools are available from the PGS Toolkit. First, a test plan will be designed, then a widely test of the shell using the PGS Toolkit will be performed. Evaluate the adequacy of the resulting implementation and report PGS Toolkit problems in a timely manner. An iterative process between SDST and PGS/ECS for revision of the shell and/or the Toolkit will continue until serious problems have been resolved.

6. Integration of the Algorithm into the Shell
   -- Check the interconnections between the algorithm and other subsystems and evaluate the flexibility of algorithm installation. The integration test will be conducted by a test plan which will be accomplished between 12/93 - 1/94. The data products, before and after integration of the algotithm, will be examined carefully. Document the interface control between the algorithm and the shell. Contact TM and PGS and correct any serious inconsistencies.

7. Discussion with TM and PGS
   -- There will be many discussions between TM, PGS, and SDST for resolving any problems and improving the system computation. The performance of the integration test will be discussed with TM and PGS. Document processing log and highlight the potential problems. Report the test results to TM, PGS/ECS, and SDST.

8. End-to-End Test
   -- The SDST will deliver the prototype of the shell to PGS and perform an end-to-end test using simulated data to confirm that the integrated system works correctly. This test will also provide an opportunity to find out the consistency between the shell development environment and the operational environment.

DRAFT



Comments:

1. The PGS Toolkit, particularly the Level 1-4 Product Access Tools, Temporary File I/O Tools, Browse Output Tools, and Quicklook Product Tools, is very important to the Shell Development. It is expected that these tools are available before April, 1993.
2. It is expected that a typical algorithm is available to test before May, 1993.
3. At the End-to-End test, the typical integration is completed and the SDST will deliver the software to the EOSDIS for testing in the operational environment.

Risk Analysis in the Level-2 Processing Shell Design and Development

Risk analysis in the shell design and development provides a systematic analysis to categorizing threats to the system implementation and proposes an appropriate approach to handle these threats. The risk analysis should result in an optimal plan of action that balances risk of failure against the cost of the additional controls.

Basically, the risk areas could cover (1) schedule, (2) hardware, (3) software, and (4) manpower required in the system integration. The identification of risk areas in the shell design and development is discussed below.

1. Schedule Risks:
    1. Based on the current EOS Science Software Development Schedule, the specifications of the PGS Toolkit will be reviewed at the 4th quarter, 1993, which could delay the shell development significantly.
       -- Action: Provide tools requirements and priority to ECS contractor at the end of 1992 and check the progress of the Toolkit development frequently.
    2. It is expected to have a typical algorithm for the integration test at the end of April, 1993.
       -- Action: Contact TM frequently for their code progress. Also, generate pseudo code in the shell development for appropriate simulation.

2. Hardware Risks:
    1. System specifications based on current technology are not efficient for huge volume data processing.
       -- Action: Keep the shell flexible for future installation in advanced hardware environment.
    2. The maintenance, operation, and security of hardware are important to avoid any unnecessary time delay.
       -- Action: Contact key persons for help.
    3. To be determined requirements are still not determined.
       -- Action: Design the shell with some assumptions and keep the shell flexible for future improvement.

3. Software Risks:
    1. The tools required for the shell development are not available when they are needed.
        -- Action: Generate pseudo code to simulate the functions of these tools during the initial development period.
    2. The PGS Toolkit delivered to the TLCF does not match the system requirements.
        -- Action: Contact PGS/ECS staff to resolve any problems.
    3. The fundamental algorithm concept could be changed which makes the time for system integration increased significantly.
        -- Action: Communicate with TM for any potential change and keep the shell flexible for any structure modification.
    4. It is very diffucult to exercise all of the viable paths in a complicated code.
        -- Action: Work with TM to design the strategy for optimizing the code test. Also check the performance for some invalid input data or meaningless data.
    5. Test data and ancillary data may be incomplete, insufficient, or incorrect.
        -- Action: Work with TM to identify the test results for further software modification. Make the shell robust for handling inappropriate data.

4. Manpower Risks:
    1. Manpower is not enough to support TM code test.
        -- Action: Maintain the manpower flexible to assist in some urgent tasks or increase manpower.
    2. The key persons are changed for unpredictable reasons.
        -- Action: Maintain a complete document of each task and record the current progress, so that the new staff can continue the task quickly.